REMARKS

This is a response to the final Office action dated October 5, 2005. Claims 1-18 are presented for examination. Applicants appreciate the grant of an interview with Examiner Gold on February 22, 2006, understand the drawings and specification are accepted, and request reexamination and reconsideration of the application.

In paragraph nos. 1-2 of the Office action, the Examiner rejects claims 1-18 under 35 USC 102(b) as being anticipated by U.S. Patent No. 6,510,439 B1 to Rangarajan et al. (Rangarajan).

In response, Rangarajan cannot anticipate claim 1 because it fails to describe a client-side caching system as recited in claim 1.

First, let's clarify the client-side and server-side of Rangarajan's system. Referring to Figure 1 and accompanying specification (e.g., col. 4, 6, and 7), Rangarajan's server-side is inside the dotted lines (e.g., HTTP server 16, CGI script 18, documents 11, state management server 12, registration table 13, and GSUI 22) while the client-side is outside the dotted lines (e.g., end user or client 24).

Rangarajan's CGI script 18 is not a client-side script. Instead, the CGI script 18 is executed only on the server side. As shown in Figure 1 and stated in col. 7, lines 6-15 of Rangarajan, a CGI script 18 is used as an interface between the HTTP server 16 and the state management server 12. When a client request is received, the HTTP server 16 sets the CGI environment variables to reflect the full URL of the requested document and the cookie(s) accompanying the client's HTTP request. The CGI script 18 is then executed. The script 18 is configured to establish an internet socket connection with the state management server 12 and then forward the URL and any received cookies to the state management server 12.

In the interview, we clarified that Rangarajan's server does not send the CGI script to the client. As indicated in col. 7, lines 30-44 of Rangarajan, the state management server 12 on the server side receives a forwarded URL and a cookie from the CGI

6

1   program (i.e., CGI script 18), determines what version of the document should be

2   provided to the client, i.e., the last version of the group that was accessed, and which

3   document in the group that was accessed. The cookie state information is then revised

4   to indicate the new reference and the file path, and a new cookie is returned to the

5   HTTP server 16 by the CGI script 18. The HTTP server 16 then retrieves and returns

6   the document and the modified cookie to the client 24. It is clear the CGI script 18

7   executes its functions on the server side. It is also clear that the HTTP server 16 does

8   not send the CGI script 18 to the client for execution.

9
    In the interview, we also clarified Rangarajan's server does not send a client-side script
10
    that appends the cookie value to the request for a resource and causes the client to
11
    automatically request the resource with the appended cookie value so that if the most
12
    recent version of the resource is in the client cache, the resource is retrieved from client
13
    cache rather than from the server, and if not, is retrieved from the server.
14

15   There cannot be anticipation of claim 1 because Rangarajan fails to describe a server

16   for sending a response including a cookie and a client-side script to the client, wherein

17   the cookie value represents the last version of the resource, and the client-side script

18   appends the cookie value to the request for a resource and the client automatically

19   requests the resource with the appended cookie value so that if the most recent version

20   of the resource is in the client cache, the resource is retrieved from client cache rather

21   than from the server, and if not, is retrieved from the server.

22   The Office action cites to Rangarajan's use of the EXPIRES field (e.g., col. 9, line 65 -

23   cpl. 10, line 11), but this is not what is set forth in claim 1.

24
    Further, an EXPIRES field (i.e., expires HTTP header) will not fully solve the problem of
25
    out-of-date pages. As stated in paragraph 0007-0008 of our application, a web server
26
    that delivers up a daily TV schedule can set the "expires" HTTP header on all of the TV
27
    schedule pages to be midnight. If a user navigates to the TV schedule page more than
28
    once on the same day, the browser doesn't need to ask the web site for a new page; it
29
    will simply display the page stored in its cache. At midnight the browser will expire the
30

7

1   old TV schedule page from the cache, and subsequent requests for the page will cause
2   the browser to once again request the page from the web server. However, if the server
3   changes the TV schedule page in the middle of the day, a person returning to the web
4   page after the change will see a cached page, which is out-of-date. Caching using the
5   expires header is difficult for a web application because the application's pages may
6   change frequently. For example, in the case of a financial management application, the
7   server may need to refresh all the pages when the user changes the background color,
8   but only banking-related pages when the user updates their bank balance. These
9   examples from the application also clarify Rangarajan fails to disclose claim 1.

10
     In view of the above, it is respectfully submitted claim 1 is allowable over Rangarajan.
11

12   As far as claims 2-18, we respectfully submit that they are also allowable based on the
13   following:

14   Claim 2 is allowable based on its dependency on allowable claim 1.
15

16   Claim 3 is allowable based on its dependency on allowable claim 1. Separately, claim 3
17   is allowable over Rangarajan, because Rangarajan fails to suggest the server response
18   includes a non-displayed relatively small page and the client-side script is in the entity
19   body of the response.

20   Claim 4 is allowable based on its dependency on allowable claim 1.
21

22   Independent claim 5 and its dependent claims 6-7 distinguish over Rangarajan for the
23   reasons presented in connection with allowable claim 1.

24   Claim 8 is allowable based on its dependency on allowable claim 1.
25

26   Independent claim 9 and its dependent claim 10 distinguish over Rangarajan for the
27   reasons presented in connection with allowable claim 1.

28   Claim 11 is allowable based on its dependency on allowable claim 9. Separately, claim
29   11 is allowable over Rangarajan, because Rangarajan fails to suggest the server
30

8

1   response is a relatively small non-displayed page and the client-side script is in the
2   entity body of the response.

3
4   Claims 12-14 are allowable based on their dependency on allowable claim 9.
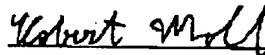
5   Independent claim 15 and its dependent claims 16-17 distinguish over Rangarajan for
6   reasons presented in connection with allowable claim 1.

7
8   Independent claim 18 distinguishes over Rangarajan for the reasons presented in
9   connection with allowable claim 1.

10 It is respectfully submitted the application is in condition for allowance.

11 Please call if you have a question, comment, or it will expedite progress of the
12 application.

13

14

15   Respectfully submitted,

16

17

18   *Robert Moll*

19   Robert Moll
20   Reg. No. 33,741

21

22

23

24 1173 Saint Charles Court
25 Los Altos, CA 94024
26 Tel: 650-567-9153
27 Fax: 650-567-9183
28 Email: rgmoll@patentplanet.com
29
30

9